8) Resume the suspended original thread (use ResumeThread) of the new process. The untrusted module's initialization code does not run until the original thread is resumed. But the initialization of the trusted module is executed as part of LoadLibrary. The net effect is to establish the necessary Bypass Protector and Tampering Protector context before any code from the untrusted module has executed.

CLAIMS

1. A computer system including an operating system and software applications, the system comprising:

a central processing unit;

means for storing and retrieving programs and data connected with said central processing unit;

an operating system stored in said means for storing and retrieving programs and data;

a plurality of software applications stored in said means for storing and retrieving programs and data;

a plurality of application threads, wherein each of said threads is associated with a single one of said software applications;

a plurality of bypass protocols that interface with said software applications, wherein each of said bypass protocols is associated with a single one of said software applications;

a bypass driver that interfaces with said bypass protocols, wherein the specific state of trust of each of said application threads of said software applications

associated with said bypass protocols is obtained by said bypass driver from said bypass protocols;

a thread trust datastore that interfaces with said bypass driver, wherein the state of trust of said software applications is communicated from said bypass driver to said thread trust datastore and stored in memory; and

a system service dispatch tap that interfaces with said operating system, wherein invocations of services from said operating system by said software applications are intercepted by said system service dispatch tap, the state of trust of said software application is obtained from said thread trust datastore, and said invocation of service is routed in said operating system based upon said state of trust.

2. The computer system including an operating system and software applications of claim 1, wherein said operating system is a Microsoft Windows® based operating system.

3. The computer system including an operating system and software applications of claim 1, wherein said central processing unit is an Intel®-based microprossor.

4. The computer system including an operating system and software applications of claim 1, wherein said invocation of service is terminated if the value of said state of trust of said software application is negative.

5. A computer system including an operating system and software applications, the system comprising:

a central processing unit;

means for storing and retrieving programs and data connected with said central processing unit;

an operating system stored in said means for storing and retrieving programs and data;

a plurality of privilege levels associated with said central processing unit;

a plurality of software applications stored in said means for storing and retrieving programs and data, wherein each of said applications is associated with a single one of said privilege levels;

a plurality of application threads, wherein each of said application threads is associated with a single one of said software applications;

a plurality of driver modules stored in said means for storing and retrieving programs and data, wherein each of said driver modules are associated with a single one of said privilege levels;

a plurality of return addresses, wherein each of said return addresses are associated with a single one of said software applications;

a plurality of driver requests, wherein each of said driver requests is associated with a single one of said software applications and a single one of said return addresses, and said software application associated with each of said driver requests is associated with a lower privilege level than the privilege level associated with the driver to which said request is directed;

a plurality of bypass protocols that interface with said software applications, wherein each of said bypass protocols is associated with a single one of said software applications and a single one of said driver modules;

a thread trust datastore that interfaces with said driver modules, wherein the return addresses of said software applications are obtained by said driver modules and stored in said trust datastore, and one of said return addresses associated with one of said software applications may subsequently be retrieved by said driver modules, compared with one of said return addresses associated with one of said driver requests from one of said software applications, and said driver request is routed differentially based on whether said return address associated with said driver request is associated in said driver module's thread trust datastore with the requesting thread's application.

6. The computer system including an operating system and software applications of claim 5, wherein said driver request is denied if said return address of said software application does not match one of said return addresses associated with said calling thread's software application in said thread trust datastore.

7. The computer system including an operating system and software applications of claim 5, wherein said operating system is a Microsoft Windows® based operating system.

8. The computer system including an operating system and software applications of claim 5, wherein said central processing unit is an Intel®-based microprossor.